Sistemas inteligentes para la detección y filtrado de correo spam: una revisión

José R. Méndez[1], Florentino Fdez-Riverola[1], Fernando Díaz[2], Juan M. Corchado[3]

[1]Departamento de Informática, Universidad de Vigo
{moncho.mendez, riverola}@uvigo.es
http://sing.ei.uvigo.es/
[2]Departamento de Informática, Universidad de Valladolid
fdiaz@infor.uva.es
http://www.infor.uva.es/~fdiaz
[3]Departamento de Informática y Automática, Universidad de Salamanca
corchado@usal.es
http://bisite.usal.es/

Resumen

Este artículo presenta una revisión general de los modelos de detección y filtrado de correo spam existentes en la actualidad. En concreto, se realiza una subdivisión de las técnicas existentes en dos grandes tipos: modelos basados en la colaboración de usuarios y modelos basados en el análisis de contenido. Se presentan las características específicas del problema y se analizan los corpus públicos disponibles, así como las técnicas habituales empleadas en su preprocesamiento. Además, se realiza una revisión de los distintos sistemas implementados destacando las características distintivas de cada uno de ellos. El trabajo finaliza con la exposición de las conclusiones más destacables acerca del estado del arte actual.

Palabras clave: spam, aproximaciones colaborativas, filtros basados en contenido, revisión.

1. Introducción y Motivación

Spam es el término que se emplea comúnmente para designar el correo electrónico no solicitado que se envía a través de Internet. Según SpamHaus [SpamHaus05a], el término spam aplicado a mensajería electrónica hace referencia a aquellos correos electrónicos enviados de forma *masiva* y que *no* han sido *solicitados* por sus destinatarios. Ambas características (masivo y no solicitado), deben estar presentes en un mensaje para que sea considerado como no legítimo.

El origen del término spam tiene poco que ver con el correo electrónico. Durante el año 1937 comenzó a comercializarse una carne enlatada denominada *Hormel's Spiced Ham*. Su consumo aumentó durante la Segunda Guerra Mundial al ser parte de la dieta diaria de los soldados norteamericanos. Debido a la fama del producto, finalmente su nombre se abrevió dando lugar al término spam (*SPiced hAM* o jamón sazonado). Posteriormente, en el año 1969, un vídeo cómico de Monty Python que ridiculizaba la presencia de esta carne en todos los platos de un restaurante, terminó estableciendo el símil entre algo molesto y repetitivo con la llegada incesante de correos no deseados.

Desde un punto de vista estrictamente técnico, el calificativo spam se aplica a todos aquellos

ISSN: 1137-3601.

mensajes en los cuales (*i*) la identidad personal del receptor y el contexto es irrelevante, puesto que el mensaje es potencialmente aplicable a un gran número de destinatarios sin importar realmente cuál sea el receptor y (*ii*) el emisor no dispone de un permiso verificable y revocable emitido por el receptor del mensaje [SpamHaus05a]. En esta definición, la idea de permiso hace referencia a que el usuario consiente, aunque sea implícitamente, la recepción de un mensaje. En [SpamHaus05a] se defiende la idea comúnmente aceptada, de que el concepto de spam está íntimamente ligado con el consentimiento de un mensaje y no con su contenido.

Estudios de ámbito internacional realizados acerca de este fenómeno en el año 2005, muestran que aproximadamente el 68% del tráfico mundial generado por correos electrónicos corresponde al envío de mensajes spam [TheRegister05]. Si se reduce el ámbito a EEUU, este porcentaje se eleva hasta alcanzar el 87%. Según un estudio de SpamHaus acerca de las direcciones origen de mensajes spam [SpamHaus05b], la mayoría de estos correos se genera en EEUU, Corea y China, a través de servicios proporcionados por PSI (*Proveedores de Servicio de Internet*) e individuos bien conocidos.

Respecto al uso de Internet en España, la AIMC (*Asociación para la Investigación de Medios de Comunicación*) publicó un estudio que cuantificaba en 12.847.000 los usuarios de Internet entre los meses de abril y mayo de 2005 [AIMC05]. En este estudio, se constata que tanto el servicio web como el correo electrónico, son las herramientas más empleadas por los usuarios de Internet en nuestro país. El primero de ellos es usado por el 95% de los usuarios (11.562.300 personas) mientras que el segundo acapara a un 83% (10.663.010 individuos). Relacionando utilización de Internet y existencia de correos spam, la AUI (*Asociación de Usuarios de Internet*) afirma que el porcentaje de mensajes spam enviados utilizando este medio aumenta de forma progresiva (del 58% en diciembre de 2003 al 64% en mayo de 2004) [AUI05].

Con el objetivo de paliar el aumento de esta práctica maliciosa, durante los últimos años se han venido adoptando de forma gradual en EEUU y Europa distintas medidas legislativas [Moustakas et al. 05]. Sin embargo, tomando como base los estudios realizados al respecto, resulta evidente que la aplicación de la legislación vigente tiene una eficacia muy limitada. En este sentido, se ha constatado que la cantidad de spam que reciben los usuarios de correo electrónico aumenta día a día, a pesar de la incorporación de leyes cada vez más estrictas y del uso de nuevos y más efectivos medios de investigación criminal. A día de hoy, la mejor forma de atajar algunos de los inconvenientes ocasionados por este fenómeno consiste en el uso de filtros software, capaces de discernir entre mensajes legítimos y spam de manera automática [González et al. 05].

A continuación se expone la organización del resto de secciones del artículo: la Sección 2 realiza un estudio sobre la complejidad inherente al problema planteado, las fuentes de información y los datos disponibles, así como las técnicas de preprocesado de los distintos corpus públicos existentes. La Sección 3 presenta una revisión del conjunto de aproximaciones empleadas para la detección de correo spam, realizando una clasificación en función del tipo de técnica empleada. Finalmente, la Sección 4 presenta un resumen del trabajo realizado, destacando las conclusiones más relevantes en este campo.

2. El Problema del Correo Spam

A menudo se ha comentado que el envío de mensajes spam representa una amenaza asimétrica ya que, técnicamente, es muy fácil generar y enviar mensajes de forma masiva, mientras que se requiere de una organización sofisticada y costes muy elevados en el destino para poder ser eliminado [Gomes et al. 05].

A este respecto, en el trabajo de Mueller [Mueller05] se ponen de manifiesto un conjunto de consideraciones por las que el spam perjudica a usuarios finales de correo electrónico. De entre ellas, destacan las siguientes:

- El spam es el único medio publicitario en el que el receptor del mensaje paga más que el emisor.
- Existen distintos tipos de mensajes spam que incorporan frases en las que el usuario debe solicitar la baja de una lista de correo electrónico a la que nunca se ha suscrito. Estas actuaciones permiten al emisario de dichos mensajes, conocido comúnmente con el nombre de *spammer*, confirmar la existencia de la dirección de correo electrónico y, de esta forma, hacer un uso más abusivo de ella.
- La mayoría de los mensajes spam anuncian

productos sin ningún valor, engañosos, y en parte o en su totalidad fraudulentos.

- Los mensajes spam siempre llegan con una lista de nombres de personas afirmando que desean recibir publicidad.
- El problema del correo spam se beneficia de la disparidad de los diferentes marcos legales de protección al consumidor que existen en los países. De este modo, se convierte en la mejor vía para promocionar productos o servicios ilegales o rechazables.

Desde un punto de vista empresarial, el problema analizado afecta también a entidades con fines comerciales, deteriorando la confianza otorgada por sus clientes y mermando la capacidad de los equipos hardware. En concreto, el problema del correo spam perjudica a empresas finales de varios modos:

- Consumiendo recursos computacionales de servidores intermedios, que quedan colapsados con las tareas de envío de mensajes spam.
- Los servidores intermedios que envían correos spam son identificados por los filtros antispam, produciéndose así el bloqueo del dominio de la empresa en Internet.
- La reputación de la empresa víctima queda comprometida y se produce una pérdida de confianza por parte del cliente.

Por otro lado, también los PSI son víctimas del correo spam, pues deteriora la calidad del servicio ofrecido a sus clientes, obligando a invertir cantidades importantes de dinero para poder paliar los efectos ocasionados. Entre otros, los principales inconvenientes del correo spam para las empresas proveedoras de servicios Internet son los siguientes:

- Los clientes son las víctimas reales del correo spam y presionan de forma continua para solventar los inconvenientes ocasionados. De esta forma, los PSI se ven obligados a invertir grandes cantidades de dinero en complejas arquitecturas software y hardware con el objetivo de eliminar los mensajes spam.
- Algunos spammers emplean la técnica *hit and run* (golpea y corre), contratando los servicios de un PSI durante unos días (a veces de forma gratuita) y enviando cientos o miles de mensajes. Una vez realizada la operación abandonan los servicios del proveedor dejando su reputación comprometida.

2.1. Complejidad Inherente al Problema

Este apartado pone de manifiesto la complejidad asociada al problema de la detección y filtrado de mensajes spam, incidiendo en la dificultad de su identificación motivada por los ataques que los spammers llevan a cabo para deteriorar los filtros existentes.

Resulta obvio que la percepción de la información contenida en un mensaje por parte de un programa informático (parte izquierda de la Figura 1) es distinta a la de un humano (parte derecha de la Figura 1) y, por lo tanto, la detección de mensajes spam resulta una tarea compleja para el software.

```
<a href="dvd.example.com">
<img src="dvd.example.com/images/e1">
</a>
```

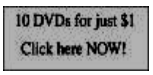

Figura 1. Percepción de un mensaje spam

A pesar del incremento en la precisión y la popularidad de los filtros antispam, los spammers encuentran nuevas formas de asegurar el envío de sus mensajes. La introducción de ruido en los correos, el cambio de las formas de envío y el desarrollo de ataques a filtros antispam, son las formas más comunes para conseguir el envío masivo de correos electrónicos [Wittel04; Lowd05]. En el trabajo de Wittel y Wu [Wittel04] se propone una clasificación de ataques a filtros antispam basada en la forma del ataque, que se detalla a continuación:

- *Generación de confusión a la hora de separar las palabras*: con este tipo de ataques el spammer consigue que la separación de los distintos vocablos que componen el mensaje, se lleve a cabo de forma errónea por el filtro antispam.
- *Perturbación del contenido*: el contenido del mensaje se oculta empleando códigos distintos, como incluir codificación de URL (*Universal Resource Locator*) en HTML, sustitución de caracteres (¡i1), codificación Base64, etc.
- *Estadísticos débiles*: intentan confundir los recuentos estadísticos de términos en un mensaje, para que el filtro experimente cierta dificultad a la hora de clasificar los correos. El uso de palabras aleatorias al final del mensaje, marcas HTML falsas o textos aleatorios extraídos de otras fuentes, son versiones comunes de estos ataques.

Corpus ID#	Dispone de Fechas	Legítimo [%]	Spam [%]	Nº total Mensajes	Formato de Almacenamiento	Autor(es)
Ling Spam	No	83,3	16,6	481	Tokens	[Androutsopoulos et al. 04]
PU1	No	56,2	43,8	1.099	Token Ids	[Androutsopoulos et al. 04]
PU2	No	80,0	20,0	721	Token Ids	[Androutsopoulos et al. 04]
PU3	No	51,0	49,0	4.139	Token Ids	[Androutsopoulos et al. 04]
PUA	No	50,0	50,0	1.142	Token Ids	[Androutsopoulos et al. 04]
SpamAssassin02	Sí	84,9	15,1	3.299	RFC 822	[Mason05]
SpamAssassin03	Sí	68,8	31,2	6.033	RFC 822	[Mason05]
Spambase	No	39,4	60,6	4.601	Feature Vectors	[Hettich et al. 98]
Junk-Email	Sí	19,3	80,7	1.563	XML	[Orasan02]
Bruce Guenter	Sí	0,0	100,0	171.000	RFC 822	[Guenter98]
Judge	Sí	0,0	100,0	782	RFC 822	[Judge02]
Divmod	Sí	0,0	100,0	1.247	RFC 822	[Divmod05]
Grant Taylor	Sí	0,0	100,0	2.400	RFC 822	[Taylor04]

Tabla 1. Información sobre los corpus disponibles

- *Estadísticos fuertes*: un ataque de este tipo se diferencia de uno débil por la naturaleza de las palabras añadidas. Empleando palabras escogidas bajo una rigurosa selección, el riesgo de que el filtro sea engañado es mayor. Un ejemplo de este ataque es el demostrado en el trabajo [Graham-Cumming04].
- *Ataques con palabras dispersas*: correspondiente a correos spam que tienen muy pocas palabras o una simple URL.
- *Intentos de ruptura de firmas*: se trata de correos spam que añaden caracteres aleatorios para atacar a filtros colaborativos, basados en comunidades de usuarios que comparten firmas de correos electrónicos spam.

En términos generales, los ataques que introducen confusión a la hora de detectar palabras así como la perturbación de contenido, se designan habitualmente con el término *ruido*. Este concepto se suele confundir con el de *concept drift*. El primero hace referencia a la dificultad de reconocer ciertos términos en el mensaje spam, mientras que el segundo conlleva una alteración del contenido semántico de los términos, motivado por el paso del tiempo [Tsymbal04].

En dominios en los cuales ocurre el fenómeno del concept drift, los modelos empleados deben ser capaces de detectar los cambios y adaptarse de forma rápida a ellos. Este fenómeno puede presentarse de forma súbita o de forma paulatina [Stanley03]. Una situación especial de concept drift se produce cuando, a pesar de no verse alterada la semántica del problema, la distribución de los datos cambia. Este tipo de manifestación se conoce con el nombre de *virtual concept drift* [Widmer93]. Normalmente virtual concept drift y concept drift aparecen de forma conjunta, como por ejemplo en el filtrado de mensajes spam [Stanley03], aunque también es posible que ocurran por separado. En la práctica, no importa el tipo de concept drift que exista, puesto que en todos los casos resultan necesarios modelos capaces de manejar y adaptarse a los cambios que se producen.

2.2. Fuentes de Información y Datos Disponibles

En este apartado se describen distintos conjuntos de datos (corpus) que están disponibles a través de Internet para su descarga y uso en la creación y prueba de filtros antispam. Desde que comenzó la investigación en el área de la detección de correo spam, varios investigadores de este campo han creado y cedido a la comunidad científica sus propios corpus, que hoy en día sirven como marco de pruebas estandarizado. En este sentido, a continuación se presentan los distintos corpus disponibles acompañados de una descripción detallada de su formato, contenido, estructura y número de mensajes que almacena cada uno de ellos. Existen diferencias en cuanto a la temática, distribución de mensajes spam y legítimos y preproceso llevado a cabo. La Tabla 1 muestra un resumen de la información disponible.

En el trabajo de [Androutsopoulos et al. 04] se puede encontrar información acerca de los corpus creados por este autor, así como las direcciones de descarga desde Internet. La principal diferencia entre estos corpus y los demás, es que se encuentran preprocesados, lo cual puede constituir una desventaja a la hora de probar nuevas aproximaciones que necesiten características que ya se han eliminado o que realicen un preproceso distinto. Además, se han eliminado todos los atributos de los correos electrónicos a excepción del

cuerpo del mensaje y su asunto (*subject*). Este investigador es autor de los siguientes corpus:

- *Ling Spam*: se trata de un conjunto de mensajes extraídos de una lista de distribución sobre la profesión y la ciencia de la lingüística. El preproceso llevado a cabo incluye la eliminación de marcas HTML y de mensajes spam duplicados recibidos en el mismo día, así como la separación en palabras (*tokens*). Cada signo de puntuación (¿, ., $, etc.) se considera un token distinto. El corpus se distribuye en 4 directorios que contienen los mismos mensajes, incorporando al preproceso anteriormente reseñado otros mecanismos adicionales como: (*i*) *bare*: no se realiza ninguna acción adicional, (*ii*) *lemm*: añade un proceso de lematización (*stemming*), (*iii*) *stop*: incluye el borrado de los términos contenidos en una lista de parada (*stop word list*), y (*iv*) *lemm_stop*: para la cual se han empleado los mecanismos usados en (*ii*) y (*iii*). Cada una de las versiones del corpus ha sido dividida en 10 partes con el objetivo de facilitar la realización de experimentos con validación cruzada (*10 fold-cross validation*) [Kohavi95].

- *PU1*: se trata de un conjunto de mensajes recibidos por un usuario particular. Para garantizar la privacidad de este usuario y de los propios correos, los mensajes se distribuyen sustituyendo cada una de las distintas palabras por un identificador único, de tal forma que no sea posible la reconstrucción del texto original pero sí la aplicación de técnicas de Inteligencia Artificial (IA). Al igual que Ling Spam, se distribuye en cuatro versiones (*bare*, *lemm*, *stop* y *lemm_stop*) con idéntica descripción y estructuradas también en 10 partes de igual tamaño. Cada mensaje se almacena en un fichero de texto comprimido mediante la herramienta gzip, conteniendo los identificadores de las palabras del cuerpo y el asunto del mensaje separados por espacios en blanco. Para distinguir los tipos de mensaje se codifican los nombres de los ficheros que contienen mensajes spam de la forma **spmsg*.txt*, frente a los que contienen mensajes legítimos con representación **legit*.txt*.

- *PU2*: el preproceso y la forma de distribución coincide con el empleado para el corpus anteriormente descrito.

- *PU3*: para recopilar correos spam se emplearon mensajes sin duplicados procedentes del corpus PU1, además de nuevos correos recopilados mediante donaciones u otros corpus como SpamAssassin. Se distribuye preprocesado y de la misma forma que el corpus PU1.

- *PUA*: incluye mensajes en varios idiomas. Se distribuye en el mismo formato y de la misma forma que los corpus PU1, PU2 y PU3.

A mayores de los corpus anteriormente comentados (creados por el grupo de Androutsopoulos), existen otras colecciones de mensajes que mayoritariamente se encuentran disponibles en el formato estándar de intercambio de correo electrónico (RFC 822). Esta norma internacional permite especificar con considerable detalle las cabeceras, el cuerpo y los archivos adjuntos de cada correo, lo que posibilita el acceso a toda la información original contenida en los mensajes que componen el corpus.

- *SpamAssassin versión 02 y versión 03*: SpamAssasin implementa un software de filtrado spam desarrollado por el grupo Apache que consulta varias redes de intercambio de firmas de mensajes spam e incorpora, en su última versión, un filtro bayesiano basado en el trabajo de Paul Graham [Graham02]. La versión del 2003 añade mensajes al corpus anterior y, por razones de privacidad, sustituye los dominios de los remitentes de los mensajes por *spamassassin.taint.org*. Los mensajes se han dividido en 5 grupos que identifican el tipo de mensaje y, en el caso de mensajes legítimos, la dificultad de clasificación correcta. Estos grupos son: (*i*) *spam*: 500 mensajes spam recibidos de fuentes habitualmente no spam, (*ii*) *easy_ham*: 2.500 mensajes legítimos fáciles de clasificar como tal, (*iii*) *hard_ham*: 250 mensajes legítimos que son parecidos en varios aspectos a los mensajes spam, (*iv*) *easy_ham_2*: 1.400 mensajes no spam que constituyen la incorporación más reciente al corpus y (*v*) *spam_2*: 1.397 mensajes spam que también se integran en la última ampliación del corpus.

- *Spambase*: el repositorio de aprendizaje automático UCI [Hettich et al. 98], aglutina un conjunto de bases de datos con información sobre la cual se pueden aplicar técnicas de aprendizaje automático. Entre estas bases de datos se puede encontrar Spambase, donde cada vector contiene las frecuencias de 57 atributos preseleccionados. Spambase es incluso más restrictivo que Ling Spam, puesto que no se puede variar ni el número ni la selección de los términos a usar por el modelo.

- *Junk-Email*: el número total de mensajes

diferentes, una vez eliminados los duplicados, es de 673. Una de las desventajas de este corpus radica en que no contiene toda la información codificada en el formato descrito por el RFC 822. La descarga del corpus se realiza empaquetada con el sofware tar y comprimida con gzip.

- *Bruce Guenter*: este investigador ha venido almacenando correos electrónicos spam desde 1998, cambiando la dirección de destino de los mensajes por *bait@em.ca*. Desde entonces ha recopilado más de 171.000 correos spam para la investigación de la conducta de los spammers y el desarrollo de nuevas técnicas de filtrado de mensajes. Dichos correos están disponibles en una sección de archivos spam de su página web personal [Guenter98]. Los mensajes recopilados se encuentran agrupados por año, en uno o varios ficheros empaquetados con tar y comprimidos con bzip2.
- *Judge*: proporciona una base de datos de correos electrónicos spam bien conocidos que se pueden usar para probar, desarrollar y comparar técnicas antispam [Judge02]. Desde su servidor FTP se pueden descargar hasta 782 mensajes spam comprimidos y codificados en el formato definido por el RFC 822.
- *Divmod*: representa una comunidad de software libre que ofrece servicios a proyectos desarrollados en el lenguaje de programación Python. En este sitio se ubica un conjunto de 1.247 mensajes spam [Divmod05] codificados según el RFC 822.
- *Grant Taylor*: compartió durante varios años aproximadamente 2.400 mensajes spam recopilados a partir de su propio correo [Taylor04].

2.3. Técnicas Habituales de Preprocesamiento

La inmensa mayoría de los corpus de prueba disponibles en Internet, se encuentran preprocesados mediante el empleo de diversas técnicas que derivan de trabajos de investigación previos en el campo de la recuperación de información y clasificación de textos.

El preprocesamiento de un mensaje de correo electrónico comienza con la extracción, a partir de su contenido, de los distintos vocablos (*tokens*) que lo componen. En textos normales, esta fase se lleva a cabo identificando los grupos de caracteres separados por signos de puntuación, espacios en blanco o combinaciones de ellos. Sin embargo, en el caso de correos electrónicos spam, el empleo de este mecanismo puede desechar información importante del mensaje, puesto que términos en los que el spammer introduce ruido (p.ej.: "v¡agra" o "v.agra") no serán bien procesados. La separación de palabras se realiza de forma habitual utilizando los siguientes criterios:

- Empleando únicamente los espacios en blanco como separadores (utilizado habitualmente por su sencillez).
- Llevando a cabo una etapa previa de procesamiento de ruido (aclaración o deofuscación) [Lee05].

En este sentido, es necesario tener en cuenta que el proceso de separación de términos puede diferir si la técnica empleada se basa en el uso de *n*-gramas. También pueden aparecer diferencias cuando se realiza la extracción de texto de ficheros adjuntos, como imágenes o ficheros con formato de documento portable (PDF, *Portable Document Format*). Una vez obtenidos los términos, y con el objetivo de uniformizar los vocablos, se realiza una conversión de todos los caracteres a mayúscula o minúscula, eliminando todos los símbolos de acentuación. Finalmente, se pueden aplicar procesos de eliminación de palabras semánticamente vacías y/o lematización. Estas técnicas de preprocesado se explican en los siguientes subapartados.

2.3.1. Exclusión de Palabras Semánticamente Vacías

Uno de los inconvenientes más importantes encontrados en el preproceso de textos, ha sido la identificación de palabras semánticamente vacías. La eliminación de estos términos, permite que las técnicas aplicadas con posterioridad únicamente tengan en cuenta aquellas palabras que aportan una mayor información semántica. Para llevar a cabo el filtrado de términos semánticamente vacíos, se emplean listas de palabras conocidas con el nombre de listas de parada (*stopword list*). Resulta simple elaborar este tipo de listas para el idioma inglés, sin embargo, para idiomas como el español este proceso se complica debido a la inexistencia de corpus y estudios estadísticos recientes sobre la lengua.

Una de las listas ampliamente utilizada para los idiomas español e inglés se distribuye de forma conjunta con el software SMART [Buckley et al. 94], construido por la Universidad de Cornell.

También se pueden emplear listas de parada que contienen expresiones compuestas por dos o más términos vacíos de significado [Allan et al. 95]. Como se ha demostrado con anterioridad [Méndez et al. 05], el empleo de listas de parada mejora la precisión de los filtros antispam.

2.3.2. Lematización

Puesto que la mayor parte de las técnicas antispam emplean las frecuencias de los términos que aparecen en los correos, la existencia de palabras derivadas de otras puede alterar los resultados de los cálculos. En consecuencia, resulta deseable reducir todos los vocablos derivados de una raíz común a un único término (lematización). Debido a este hecho, un proceso de extracción de raíces léxicas (*stemming*) podría implicar una mejora en los resultados obtenidos. En la realidad, la lematización automática es muy compleja y se sustituye por un proceso de *stripping*, que implica la eliminación de los prefijos y sufijos de cada término.

En el idioma inglés, la ausencia de género en las palabras y otras características morfológicas del lenguaje simplifican esta tarea. En esta lengua, se emplean habitualmente técnicas basadas en el algoritmo de Porter [Porter80]. Para el inglés, la aplicación de este tipo de algoritmos resulta muy fácil, obteniéndose buenos resultados con listados de sufijos como el propuesto por Rijsbergen [Rijsbergen79], con un total de 250 elementos. En otras lenguas como el castellano, el stripping puede no resultar suficiente, ya que los sufijos no se unen simplemente a una determinada raíz, sino que en dicho proceso, la raíz puede sufrir modificaciones importantes. Otra problemática asociada al castellano viene dada por la abundancia de verbos irregulares.

Como norma general, se puede afirmar que la realización de stemming no resulta aconsejable en entornos spam por la gran cantidad de ruido presente [Méndez et al. 05], aunque existen técnicas no basadas en sufijos capaces de reducir el ruido [Ahmed04].

2.3.3. N-gramas

El uso de *n*-gramas constituye un enfoque radicalmente distinto a los basados en el estudio de los términos. Si se emplea esta alternativa, será necesario realizar un proceso diferente de separación de los términos.

Básicamente, un *n*-grama es una ventana de *n* caracteres de tamaño que se va desplazando a través de todo el texto de cada mensaje. Así, un texto que consiste en la palabra " spammer " descompuesto en *n*-gramas donde *n*=3 produciría los trigramas: " sp", "spa", "pam", "ame", "mer" y "er ". Los *n*-gramas obtenidos pueden tratarse de la misma forma que los términos originales, considerando cada *n*-grama como un vocablo. La ventaja de los *n*-gramas es que permiten obviar problemas como los errores tipográficos, que se encuentran frecuentemente en documentos procesados mediante reconocimiento óptico de caracteres. De la misma forma, los *n*-gramas deberían permitir abordar con éxito la problemática de palabras con la misma raíz pero distintos sufijos, sin necesidad de llevar a cabo un proceso de lematización. El tamaño de los *n*-gramas es crucial y puede incidir de forma directa en la efectividad. Este tamaño se suele fijar de forma experimental, siendo los valores más frecuentes de *n*=3 [Smeaton et al. 94], *n*=4 [Cavnar94] o *n*=5 [Grossman et al. 95].

Otra forma de aplicar *n*-gramas consiste en su utilización a nivel de términos, resultando muy efectiva en el campo de la investigación antispam [Androutsopoulos et al. 04]. Mediante el empleo de esta variante, la frase "spam is a very hard problem" se transformaría en los trigramas "spam is a", "is a very", "a very hard" y "very hard problem". Bajo este supuesto, los modelos basados en el empleo de frecuencias de términos representativos, se podrían considerar casos particulares de modelos que trabajan sobre *n*-gramas, donde *n*=1.

3. Modelos para la Detección de Correo Spam

En esta sección se realiza una revisión del conjunto de aproximaciones empleadas para la detección de correo spam. Las técnicas analizadas están basadas en distintas heurísticas o en la observación concreta de atributos extraídos de los correos electrónicos. Inicialmente se presentan técnicas fundamentadas en la cooperación de usuarios a través de la divulgación del conocimiento sobre mensajes spam. A continuación se revisan los modelos basados en contenido, que aplican distintas técnicas supervisadas de aprendizaje automático para la detección de correo no legítimo. Finalmente, se identifican modelos creados a partir de la combinación de distintas estrategias.

3.1. Aproximaciones Colaborativas

En este tipo de técnicas normalmente no se considera el contenido del mensaje, y la clasificación de los correos se lleva a cabo mediante la colaboración de grupos de usuarios que comparten información sobre los mensajes spam. Cuando un usuario recibe un correo spam, éste comparte con el resto de usuarios uno o más datos de identidad del mensaje. El resto de usuarios que reciben el mismo correo electrónico pueden identificarlo mediante los datos proporcionados por el primer usuario.

Los identificadores compartidos normalmente son la dirección o el dominio del remitente y las firmas del contenido (*hash* o *digests*) del mensaje. Con ellos se confeccionan listas negras, de tal forma que ningún mensaje que coincida con estos criterios será aceptado.

Los siguientes subapartados presentan varias técnicas colaborativas centradas en distintos datos extraídos de los mensajes, así como diferentes modelos de cooperación entre usuarios.

3.1.1. Listas Negras y Blancas

Las listas negras y reglas sobre dominios, redes y hosts fue uno de los primeros métodos utilizados para detectar correo spam. Se basan en el uso de reglas simples de exclusión de mensajes que han sido manipulados o provienen de ciertos dominios, redes o servidores de Internet. Con estas reglas se pueden identificar y clasificar grandes cantidades de correo no deseado, sin embargo, no resulta complicado falsificar y manipular este tipo de información. Existen aproximaciones similares a las reglas, como el análisis de los logs de servidores de correo electrónico [Clayton05], el análisis de los servidores de correo por los que ha viajado el mensaje [Leiba et al. 05] o el análisis de la reputación de redes [Golbeck04].

Las listas negras de direcciones de spammers son accesibles mediante servicios web o ficheros compartidos a través de los cuales, y de forma remota, se puede consultar si una determinada dirección de correo es remitente de mensajes spam. Algunas de estas listas se encuentran compartidas en forma de ficheros de texto, que contienen remitentes o expresiones regulares acerca de direcciones de correo electrónico. En el trabajo de [DesElms05] se ofrece una amplia recopilación de sitios web que incorporan este tipo de recursos, entre los que destacan SBL [SpamHaus05b] y MAPS [TrendMicroInc05].

Como contrapartida, una lista blanca contiene una enumeración de equipos de los que nunca se debe desconfiar, y que mantienen una relación de confianza garantizada a partir de una comunicación anterior [SpammerX05]. Cada vez que el servidor de correo electrónico recibe un mensaje de un usuario en el que no confía, le envía al remitente un correo en el que le indica que para verificar su identidad, debe seguir un enlace web. Cuando el remitente accede al enlace, el correo electrónico enviado se entrega al destinatario y se establece la relación de confianza. Una vez establecida dicha relación, no se volverá a realizar la verificación de ningún mensaje posterior. Pese a su simplicidad, este tipo de mecanismos son muy difíciles de burlar, aunque se puede perder cierta cantidad de mensajes debido a personas que no desean ser verificadas. En este sentido, un inconveniente a mayores es que los mensajes generados de forma automática, como los boletines de suscripción o los procesos de confirmación de suscripción a listas o servicios electrónicos nunca serán verificados. Ejemplos de este tipo de recursos son SpamBlocked [Frostfyre05] o el programa de acreditación de remitentes de correo electrónico de Surety Mail [SuretyMail05].

3.1.2. Resúmenes

Razor [Prakash05], Pyzor [Pyzor05] y DCC (*Distributed Checksum Clearing-house*) [Rhyolite00] son aplicaciones cuyo funcionamiento se basa en el resumen de correos electrónicos no deseados. Debido a que el valor de las funciones hash habituales (como MD5) puede ser alterado fácilmente modificando un único carácter, Razor y Pyzor incorporan un algoritmo diseñado para eludir pequeñas modificaciones de caracteres y analizar el mensaje por completo. De esta forma, se eliminan variantes triviales como los números aleatorios contenidos en el asunto. Este algoritmo se conoce con el nombre de Nilsimsa [Cmelax05]. Una ventaja añadida a estas aplicaciones es la posibilidad de realizar sumas de control segmentadas, permitiendo así que el filtro de correo sólo se centre en las últimas diez líneas del correo o en las primeras cinco líneas. De esta forma, cada mensaje ha de ser totalmente aleatorio para poder evadir este tipo de filtros.

3.2. Modelos Basados en Contenido

En contraposición con las aproximaciones colaborativas, las técnicas basadas en contenido emplean características extraídas de la cabecera o del cuerpo del mensaje para realizar la clasificación de un correo. En este campo, las técnicas de aprendizaje automático gozan de un interés merecido por su habilidad probada en la clasificación de textos [Jackson02].

Los modelos basados en contenido tratan de determinar los atributos comunes a los mensajes spam y legítimos, a partir de una representación en forma de vector de las características de cada correo. Para extraer esta información de un texto, se selecciona una lista de palabras representativas de la legitimidad de los mensajes. Cada mensaje se representa con un vector de números reales o de valores lógicos que contiene, en cada posición, la frecuencia o presencia de los términos seleccionados [Androutsopoulos et al. 04]. El tamaño de la lista de términos a considerar se establece a priori, determinando la dimensión del vector que representa a cada correo y el número de atributos con los que trabajará el modelo de aprendizaje seleccionado.

La elección de los términos más representativos de cada mensaje se realiza empleando técnicas de selección de características. La técnica más habitual se basa en el cálculo de la ganancia de información (IG, *Information Gain*) de cada término con respecto a los posibles valores del atributo a predecir (legítimo o spam). El valor de IG se calcula para cada término del corpus y, finalmente, se seleccionan aquellos términos para los cuales su valor es mayor [Zhang et al. 04].

Otras dos técnicas comúnmente utilizadas para calcular el grado de representatividad de un término dentro de un conjunto de correos son la información mutua (MI, *Mutual Information*) y el estadístico χ^2 (chi cuadrado). Por su sencillez de cálculo, cabe destacar también la frecuencia de documentos que contienen un término dado (DF, *Document Frecuency*), que ha sido usada satisfactoriamente como métrica de relevancia [Zhang et al. 04].

3.2.1. Naïve y Flexible Bayes

Naïve Bayes es el algoritmo más conocido propuesto para la clasificación de textos mediante modelos de aprendizaje automático. En el ámbito del filtrado de correos spam, los filtros con base teórica sustentada en el teorema de Bayes han sido las primeras propuestas encabezadas por el trabajo de Graham [Graham02]. En los últimos años, este tipo de modelos ha adquirido una gran popularidad debido a su capacidad de representar de forma eficiente, distribuciones complejas de probabilidad. A pesar de que asumir la independencia de los atributos de un correo electrónico constituye una simplificación no realista, estudios realizados en el campo del filtrado antispam demuestran una gran efectividad [Androutsopoulos et al. 00a; Androutsopoulos et al. 00b; Androutsopoulos et al. 04]. Este tipo de clasificadores ha sido probado también utilizando atributos de posición de palabras en lugar de atributos de frecuencias, obteniendo incluso mejores resultados [Hovold05].

Flexible Bayes [John95] es una alternativa a Naïve Bayes para mensajes cuyos atributos se han representado de forma continua (frecuencias de términos). En lugar de emplear una distribución normal simple, las probabilidades de cada atributo se estiman por la media de distribuciones normales (las mismas para cada categoría), con distinta media y desviación típica común.

3.2.2. Máquinas de Vectores de Soporte

Las máquinas de vectores de soporte (SVM, *Support Vector Machines*) constituyen una familia de algoritmos ampliamente empleados en tareas de clasificación y regresión [Vapnik99]. Este tipo de modelos dispone de una base teórica sólida, que haciendo uso de la teoría del aprendizaje estadístico, garantiza un buen nivel de generalización a partir de los datos de entrada. Su funcionamiento se basa en la idea de transformar los datos existentes (espacio de características inicial) para encontrar un hiperplano de mayor dimensión donde maximizar la separación existente entre las clases. El mecanismo de aprendizaje está basado en la idea de minimización de riego estructural (SRM, *Structural Risk Minimization*) [Vapnik99].

Las SVM son especialmente apropiadas para problemas de categorización de texto y han sido utilizadas con éxito en el ámbito de la detección y filtrado de correos spam [Joachims98]. Utilizando SVM no es necesario realizar una selección de términos previa como en otros algoritmos de aprendizaje automático, puesto que su capacidad de aprendizaje no se degrada a medida que se incorporan nuevas características.

3.2.3. Boosting de Árboles de Decisión

Los algoritmos de boosting son técnicas basadas en el uso de mecanismos de aprendizaje débiles, es decir, algoritmos que aprenden con una tasa de error menor que 50%. Los árboles de clasificación C4.5 [Quinlan93] y Decision Stumps [Wayne92] (un árbol de decisión que sólo tiene el nodo raíz) se emplean a menudo con este tipo de modelo.

El funcionamiento de esta técnica se basa en combinar las hipótesis débiles generadas por los mecanismos de aprendizaje débiles, en una única hipótesis de gran precisión. Para la obtención de hipótesis diferentes, este tipo de algoritmos se ejecutan un cierto número de veces sobre distintos subconjuntos de entrenamiento. En el dominio de filtrado de correo spam, el algoritmo más empleado ha sido AdaBoost [Freund97] combinado con árboles C4.5 y Decision Stumps.

3.2.4. Ripper y Rocchio

RIPPER (*Repeated Incremental Pruning Produce Error Reduction*) [Cohen95] es capaz de inducir reglas de clasificación a partir de un conjunto de ejemplos, es decir, simplemente construye reglas de la forma *si-sino* empleando operadores de conjunción y disyunción. Se trata de una extensión de otro algoritmo previo denominado IREP (*Incremental Reduced Error Pruning*) [Fürnkranz94].

En el trabajo de Provost [Provost99] se realiza una comparativa entre Naïve Bayes y Ripper para el campo del filtrado de correo spam, obteniéndose con ambas aproximaciones tasas de error muy similares.

Cohen y Singer [Cohen99] destacan la precisión de Ripper en la clasificación de textos, a la vez que se realiza una comparación exhaustiva entre esta técnica y el modelo Rocchio. Rocchio [Rocchio71] emplea representaciones vectoriales de los documentos, de tal forma que los vectores de dos documentos con contenido similar sean semejantes. En el trabajo de Joachims [Joachims97] se pueden encontrar comparativas de Rocchio con otras aproximaciones que emplean aprendizaje automático para el problema de la clasificación de textos.

3.2.5 Chung Kwei

Chung-Kwei [Rigoutsos04] es un modelo de detección de correo spam basado en el análisis de mensajes spam y la identificación automática de patrones. Las principales ventajas de este algoritmo radican en su rapidez y capacidad para adquirir conocimiento de forma incremental.

El proceso se basa en la ejecución de un algoritmo llamado Teiresias [Rigoutsos98], capaz de encontrar patrones que aparecen dos o más veces en el corpus de entrenamiento. Una vez entrenado el modelo y ante un nuevo mensaje, éste será considerado como spam (con un grado de fiabilidad), en función del número de patrones identificados en el conjunto de entrenamiento que encajen con él.

El algoritmo Teiresias ha sido empleado satisfactoriamente en gran variedad de problemas de la ciencia [Rigoutsos98] y la seguridad informática [Feng et al. 03]. Opera en dos fases: exploración y empaquetado. Durante la primera fase se identifican los patrones con los que encaja mayor cantidad de texto. Estos patrones elementales constituyen las construcciones de bloque para la fase de empaquetado, combinándose progresivamente en patrones cada vez más largos hasta encontrar los patrones de mayor longitud. Además, el orden en el cual se realizan los empaquetados, facilita las tareas de identificación y descarte de patrones que no sean máximos.

3.2.6 Bosques Aleatorios

Los Random Forests [Breiman01] son modelos que consisten en una colección de árboles de decisión, de forma que cada árbol se construye a partir de un vector de correos seleccionado de forma aleatoria. Una vez construido el modelo y ante un nuevo mensaje, cada árbol de decisión emite un voto unitario.

Para clasificar un nuevo correo dado por un vector de características, se presenta el vector de características a cada árbol de decisión. Cada árbol emite un voto y se considera la clase que obtenga un mayor número de votos. Existen varias versiones de bosques aleatorios dependiendo de la forma en la que se genera el conjunto de entrenamiento.

En los bosques aleatorios, los parámetros fortaleza y correlación se utilizan como medida para poder establecer un límite superior para el error de generalización. La interacción entre estos dos

parámetros, proporciona la base para entender el funcionamiento de este tipo de algoritmos. En este sentido, hay dos factores que influyen en la tasa de error de generalización:

- Incrementando la correlación se incrementa la tasa del error.
- Un árbol con una tasa de error baja es un clasificador fuerte. Si se incrementa la fortaleza de los clasificadores individuales, se merma la tasa de error en el bosque.

Una reducción o aumento del número de atributos seleccionado inicialmente, afecta de forma directamente proporcional a la correlación y a la fortaleza. Éste es el único parámetro sensible en este tipo de modelos, y para su ajuste se suele emplear la tasa de error. Experimentos realizados con esta familia de clasificadores, han demostrado obtener un alto grado de precisión al ser empleados como mecanismos de clasificación en el dominio del filtrado de mensajes spam [Rios04].

3.2.7 Indexado por Semántica Latente

El indexado mediante semántica latente (LSI, *Latent Semantic Indexing*) [Deerwester et al. 90] proporciona una base teórica y un método para la extracción y representación de conocimiento. El método está basado en la utilización de información contextual de los términos en un corpus de gran tamaño, para extraer y representar el significado de las palabras y conjuntos de palabras empleando cálculos estadísticos. LSI surgió como una herramienta para la indexación y recuperación automática de información, con el propósito de superar el problema de la semántica, deficiencia que presentaban muchas de las técnicas de recuperación.

LSI construye una matriz de términos por documentos para un corpus dado, donde una posición determinada representa la frecuencia con la que aparece el término en el documento. Una vez construida la matriz, se puede realizar un preproceso empleando pesos. Normalmente se utiliza como peso local el logaritmo y como peso global la entropía [Jauregi04]. Los términos representados en la matriz son elegidos por algún mecanismo de selección de características a partir del conjunto de entrenamiento.

LSI se puede aplicar en el campo del filtrado de mensajes spam [Gee03], como mecanismo de búsqueda de los k vecinos más próximos a un documento dado, o empleando un proceso de votación con todos los correos que superan un umbral de similitud determinado.

3.2.8. ECUE

ECUE (*Email Classification Using Examples*) [Delany et al. 04] es un modelo para la detección de correo spam que hace uso de un sistema de razonamiento basado en casos (CBR, *Case Based Reasoning*). Este modelo incorpora una novedosa estrategia para hacer frente al problema del concept drift, basada en la edición del repositorio de conocimiento [Delany04].

Previo a la generación del modelo, se realiza una selección de las 700 características más relevantes empleando IG. En la fase de recuperación de casos se utiliza un algoritmo k-NN, con k=3, que permite la selección de los mensajes más parecidos al nuevo correo, y que serán empleados para elaborar una solución. Con el fin de mejorar la eficiencia del modelo, se utiliza una estructura de memoria denominada CRN (*Case Retrieval Networks*) [Lenz96], que permite una recuperación eficiente y flexible utilizando proyecciones de las características del mensaje a clasificar sobre los nodos de la red CRN. Para llevar a cabo la fase de recuperación, ECUE emplea la técnica de voto unánime, que clasifica como spam al nuevo mensaje únicamente si todos los vecinos más próximos recuperados también lo son.

Como se comentó anteriormente, el modelo realiza una actualización continua de la base de casos para combatir el problema del concept drift. En concreto, los autores proponen dos técnicas para llevar a cabo la edición de la base de casos: (*i*) BBNR (*Blame Based Noise Reduction*) y (*ii*) CRR (*Conservative Redundancy Reduction*) [Delany04].

Con anterioridad a este modelo, P. Cunningham desarrolló otra aproximación basada en casos que, sin incorporar las técnicas de edición de conocimiento comentadas previamente, conseguía mejores resultados que otras aproximaciones en presencia de concept drift [Cunningham et al. 03]. La principal diferencia de esta primera aproximación con el modelo final (ECUE), radica en el mecanismo de selección de características. En ECUE se emplea ganancia de información, mientras que en el anterior se había definido una estrategia basada en el cálculo de probabilidades OR (*Odds Ratio*).

3.2.9 Casos de Similitud

En el trabajo de Kinley [Kinley05] se presenta un sistema CBR para la clasificación de correos spam basado en el uso de casos de similitud. Los casos de similitud (*similarity cases*) agrupan varios mensajes similares en un mismo caso, y su representación se realiza empleando las características comunes y más relevantes de los casos agrupados.

La representación de la información se lleva a cabo mediante el uso de los términos presentes en el cuerpo del mensaje. Sobre los términos extraídos de los mensajes se aplican técnicas de stemming y borrado de palabras semánticamente vacías. Cada mensaje se concibe como un vector de identificadores de términos de longitud variable.

Durante el proceso de entrenamiento, para cada mensaje se calcula la cardinalidad de la intersección del conjunto de términos de cada mensaje almacenado en la base de casos con el conjunto de términos del mensaje en cuestión. El resultado se pondera según la relevancia y la frecuencia de los términos que forman parte del conjunto intersección. Finalmente, el sistema establece y almacena una relación de similitud entre el mensaje y los casos que hayan obtenido mayor puntuación.

Cuando se presenta un nuevo mensaje al sistema, éste calcula los casos de similitud a los que podría pertenecer. A continuación, para cada caso de similitud recuperado, se determina la probabilidad de que el mensaje pertenezca a dicho caso. Finalmente, se reutilizan los casos de similitud cuya probabilidad de contener al nuevo mensaje supera un determinado umbral.

A pesar de la consecución de resultados interesantes, el trabajo de Kinley no pretende desarrollar un software de filtrado comercial y, de hecho, manifiesta que los resultados obtenidos por el sistema propuesto no alcanzan la precisión obtenida por los filtros bayesianos.

3.2.10 SpamHunting

SpamHunting implementa un novedoso sistema de razonamiento basado en instancias (IBR, *Instance Based Reasoning*) propuesto recientemente para la detección y filtrado de mensajes spam [Fdez-Riverola et al. 05]. Este modelo saca partido de una representación disjunta de los mensajes, que le permite adaptarse al concept drift y obtener una menor tasa de error [Fdez-Riverola et al. 07]. SpamHunting está formado por una red de indexado de instancias similares (EIRN, *Enhanced Instance Retrieval Network*) junto con una técnica de reutilización basada en votación unánime [Fdez-Riverola et al. 06]. Con la finalidad de garantizar una adaptación continua del sistema, los autores incorporan un mecanismo de selección de características dinámico.

El preprocesamiento de correos electrónicos en SpamHunting incluye un proceso de reconocimiento de palabras basado en la utilización de espacios en blanco (como separadores de términos), así como la utilización de una técnica de eliminación de palabras semánticamente vacías. En cuanto a la representación de cada instancia, se realiza en base a atributos extraídos de la cabecera del mensaje, junto con un descriptor que incluye los términos más relevantes de cada correo. En este sentido, la selección de características empleada por SpamHunting se basa en la búsqueda de los términos más representativos de cada mensaje.

Desde la primera versión de SpamHunting, se han realizado sucesivas revisiones del proceso de selección de términos. En este sentido, en [Méndez et al. 06] se propone una forma de integrar conceptos provenientes de la teoría de la información para mejorar los resultados obtenidos mediante la ejecución del modelo.

3.3. Combinación de Aproximaciones

La mayor parte del software antispam de tipo comercial disponible en la actualidad, se ha construido empleando una combinación de algunos modelos y técnicas descritos anteriormente.

SpamAssassin [Mason05] es un filtro distribuido bajo la licencia GPL (*GNU Public License*) que combina inspección de los datos de red contenidos en las cabeceras de los mensajes, búsqueda de palabras claves que pueden determinar de forma unívoca spam, listas negras, análisis bayesiano y resúmenes de contenido (Vipul's Razor, Pyzor y DCC).

Spamto [Albrecht et al. 05] es un filtro antispam extensible basado en componentes acoplables (*plugins*). Incorpora un filtro bayesiano, un modelo basado en reglas (*Ruleminator*), análisis de dominios empleando peticiones a Google (*Domainator*) y comprobación de resúmenes Razor y Earl Grey.

Filtron [Michelakis et al. 04] es una aplicación construida para clasificar mensajes mediante un modelo de aprendizaje automático seleccionable por el usuario. En concreto, el usuario puede elegir entre los modelos Naïve Bayes, Flexible Bayes, LogitBoost (un algoritmo de boosting con árboles de decisión similar a AdaBoost) y SVM.

SpamGuru [Segal04] es una arquitectura diseñada para la construcción de un software antispam basado en un clasificador combinado que incluye: verificador de desafíos, análisis DNS, listas blancas y negras específicas de cada usuario y globales, Chung-Kwei y otras utilidades de correo electrónico.

Finalmente, Anti-Spam Gaunlet [Leiba04] propone un conjunto de capas para llevar a cabo diversos mecanismos de filtrado, con el fin de conseguir una reducción en la cantidad de correo electrónico spam recibido.

Modelo	Autor	Selección de características	Representación del conocimiento	Tipo	concept drift	Atributos	Mecanismo de aprendizaje
SBL	SpamHaus	-	Direcciones IP	Colaborativo	-	IP de servidores de correo	No dispone
MAPS	Trend Micro Incorporated	-	Direcciones IP	Lista negra	-	IP de servidores de correo	No dispone
Listas blancas	-	-	Direcciones de correo electrónico	Colaborativo	-	Direcciones de correo	No dispone
Razor	Prakash	-	Resúmenes de correos	Lista negra	-	Cuerpo y asunto	No dispone
Pyzor	Pyzor	-	Resúmenes de correos	Colaborativo	-	Cuerpo y asunto	No dispone
DCC	Rhyolite Software	-	Resúmenes de correos	Colaborativo	-	Cuerpo y asunto	No dispone
Naïve Bayes	Thomas Bayes	Previa al aprendizaje	Probabilidades	Resúmenes	-	Cuerpo y asunto	Aprendizaje previo (entrenamiento)
Flexible Bayes	John y Langley	Previa al aprendizaje	Probabilidades	Colaborativo	No	Cuerpo y asunto	Aprendizaje previo (entrenamiento)
Adaboost	Feund y Schapire	Previa al aprendizaje	Clasificadores débiles	Resúmenes	No	Cuerpo y asunto	Aprendizaje previo (entrenamiento)
SVM	Vapnik	Previa al aprendizaje	Recta que separa linealmente los datos y transformación no lineal a aplicar sobre el espacio de entrada	Colaborativo	Sí	Cuerpo y asunto	Aprendizaje previo (entrenamiento)
RIPPER	Cohen	Previa al aprendizaje	Reglas	Resúmenes	No	Cuerpo y asunto	Aprendizaje previo (entrenamiento)
Rocchio	Rocchio	Previa al aprendizaje	Información sobre frecuencias de documento y de términos	Basada en contenido	No	Cuerpo y asunto	Aprendizaje previo (entrenamiento)
Chung Kwei	Rigoutsos y Huynh	-	Patrones de mensajes spam	Basada en contenido	No	Cuerpo y asunto	Aprendizaje previo (entrenamiento)
Random Forests	Breiman	Previa al aprendizaje	Clasificadores débiles	Basada en contenido	No	Cuerpo y asunto	Aprendizaje previo (entrenamiento)
LSI	Deerwester et al.	Previa al aprendizaje	Vectores prototipo de cada documento.	Basada en contenido	No	Cuerpo y asunto	Aprendizaje previo (entrenamiento)
ECUE	Delany et al.	700 características IG	CRN con los correos	Basada en contenido	Sí	Cuerpo y asunto	Aprendizaje continuo
Casos de similitud	Kinley	-	Vectores con los términos de cada documento	Basada en contenido	No	Cuerpo del mensaje	Aprendizaje previo (entrenamiento)
Spam Hunting	Fdez-Riverola et al.	Palabras más relevantes de cada mensaje	Vectores compuestos por información extraída de la cabeza y el cuerpo del mensaje	Basada en contenido	Sí	Cuerpo y asunto	Aprendizaje continuo

Tabla 2. Caracterización de los modelos de detección y filtrado de correo spam

4. Conclusiones

En este trabajo se han introducido los conceptos básicos sobre la problemática asociada a la presencia de correo spam. En concreto, se han revisado las técnicas de IA empleadas en la actualidad para la detección y filtrado de mensajes no legítimos. Algunos de estos mecanismos como Naïve Bayes, SVM o indexado por semántica latente habían sido empleados con anterioridad de forma satisfactoria para la resolución de problemas de clasificación o búsqueda de textos. Atendiendo a la forma de funcionamiento, las diferentes soluciones analizadas se pueden clasificar en tres grandes grupos: (*i*) modelos colaborativos, (*ii*) modelos basados en contenido y (*iii*) combinación de aproximaciones. A este respecto, la Tabla 2 presenta una caracterización, a modo de resumen, de los principales modelos analizados.

Por un lado, los filtros colaborativos permiten emplear datos que puedan identificar unívocamente a cada uno de los mensaje spam recibidos por un usuario (como firmas MD5 del contenido, direcciones de origen de los correos spam o simplemente asuntos de mensajes). Estos datos identificativos son compartidos a través de una red de comunicaciones con otros usuarios de una comunidad de filtrado colaborativo. De esta forma, cuando un usuario clasifica un mensaje como spam, todos los miembros de la comunidad se beneficien del trabajo de clasificación del primero. La principal ventaja que aportan estos modelos es la ausencia de falsos positivos. Sin embargo, su reducida capacidad para generalizar sobre el conocimiento manejado, posibilita que puedan ser evitados fácilmente. Consecuentemente, este tipo de filtros posee una efectividad muy reducida. En concreto, las diferentes aproximaciones existentes pueden ser agrupadas en las siguientes categorías según el tipo de datos compartido:

- *Filtros basados en resúmenes del contenido*: se basan en la compartición de firmas (MD5, SHA1, etc.) del contenido del mensaje. Ejemplos de este tipo de filtros son Razor, Pyzor y DCC.
- *Filtros basados en datos extraídos de la cabecera*: se basan en compartir datos del mensaje como el asunto o el remitente. Dentro de esta categoría se distinguen dos tipos: (*i*) los basados en listas blancas y (*ii*) los basados en listas negras. Los primeros hacen referencia a que todos los mensajes que encajen con la identificación generada se considerarán como legítimos, mientras que los segundos hacen referencia a que serán considerados como spam. Ejemplos bien conocidos de listas negras son SBL y MAPS. Con respecto a las listas blancas, destacan SpamBlocked junto con el programa de acreditación de remitentes de correo electrónico de Surety Mail. Obviamente, este tipo de listas no tienen por que tener siempre naturaleza pública, de hecho, cualquier usuario de Internet podría construir una lista blanca o negra y emplearla de forma privada.

Por otro lado, los modelos basados en contenido analizan el cuerpo del mensaje para encontrar características que permitan diferenciar entre correos spam y legítimos. Estas características se emplean para la detección de futuros mensajes de correo electrónico. Algunos de estos modelos presentan similitudes, en cuanto a los mecanismos que emplean para realizar la extracción y representación del conocimiento. De esta forma los modelos se pueden agrupar, estableciendo la siguiente clasificación:

- *Filtros Bayesianos*: se basan en el teorema de Bayes para extraer la información asociada y emplean distribuciones de probabilidad para representar el conocimiento. Entre ellos se incluyen Naïve Bayes y Flexible Bayes.
- *Combinación de clasificadores débiles*: emplean varios clasificadores débiles (generalmente árboles de decisión) para la representación del conocimiento y son capaces de realizar clasificaciones de gran precisión mediante la combinación de estimaciones obtenidas por clasificadores débiles. Este apartado agrupa a los modelos AdaBoost y Random Forests.
- *Deducción de reglas*: el conocimiento se almacena en forma de reglas extraídas de un conjunto de correos de entrenamiento. El modelo RIPPER y su predecesor IREP se encuadran en esta categoría.
- *Identificación de patrones*: se realiza examinando exhaustivamente el cuerpo de los mensajes para tratar de deducir patrones comunes entre mensajes spam. Dichos patrones constituyen la forma de almacenar el conocimiento. El máximo exponente de este tipo de sistemas es el modelo Chung Kwei.
- *Espacios vectoriales euclídeos*: se basan en la aplicación de transformaciones sobre el espacio de características de entrada, con el objetivo de

obtener un nuevo espacio, donde se pueda establecer una separación lineal entre las categorías spam y legítimo. SVM emplea este mecanismo para la adquisición del conocimiento que se representa la identificación de un hiperplano capaz de separar las clases spam y legítimo. A nivel teórico, LSI se puede ver como una SVM modificada donde, sobre un espacio de entrada, se realiza una descomposición de valores y una reducción de la dimensionalidad (transformaciones en un espacio vectorial). La principal diferencia de este modelo con respecto a SVM se encuentra en la forma de representar el conocimiento. En el caso de LSI, se almacena mediante vectores que representan la estructura de uso de las palabras para cada mensaje del corpus de entrenamiento.

- *Análisis de frecuencia de términos y de documentos*: se basan en el estudio de frecuencias de documentos que contienen un término y frecuencias de términos sobre un corpus de entrenamiento. Los resultados de este análisis constituyen el activo principal del conocimiento almacenado por el modelo. El clasificador Rocchio utiliza esta aproximación.
- *Sistemas de razonamiento basado en casos*: se trata de modelos híbridos que se fundamentan en el uso de conocimiento adquirido en experiencias pasadas para resolver nuevos problemas. Estos modelos implementan cuatro fases cada vez que es necesario realizar una nueva clasificación: recuperación, reutilización, revisión y aprendizaje. Los sistemas ECUE, clasificación basada en casos de similitud y SpamHunting están inspirados en esta idea.

En general, se puede afirmar que los modelos analizados, salvo ECUE y SpamHunting, no emplean ninguna técnica capaz de hacer frente al problema del concept drift. El mayor inconveniente derivado de este hecho es que rápidamente pierden precisión, necesitando en la mayoría de los casos una reconstrucción completa del modelo.

Para abordar de forma efectiva el problema planteado, se hace necesaria la utilización de mecanismos de predicción capaces de realizar adaptaciones locales, con el fin de ofrecer una predicción precisa y adaptada a cada situación en particular. La información necesaria para realizar una predicción inicial (clasificación), se debe filtrar y adaptar para así, garantizar su validez. No obstante, es necesario un mecanismo de validación de resultados que sea capaz de identificar situaciones inconsistentes, en las cuales no es posible realizar una predicción fiable.

En este sentido, los sistemas CBR presentan una jerarquía de aprendizaje donde, en el nivel más simple, son capaces de actualizar la base de casos cada vez que un nuevo mensaje está disponible. La ventaja de un sistema con esta capacidad consiste en que, de forma contraria a otros modelos, no es necesario reconstruir el modelo para incorporar nuevo conocimiento. En un segundo nivel de aprendizaje, los sistemas CBR pueden realizar una reselección de las características que puedan ser más idóneas para detectar los mensajes spam. Este segundo nivel se puede aplicar regularmente sobre los nuevos datos, a medida que vayan estando disponibles. Finalmente, en el nivel más alto, los sistemas CBR pueden añadir nuevas técnicas para la extracción de características de los mensajes.

Una ventaja añadida de los sistemas CBR, es que permiten compartir con otros usuarios de Internet los casos almacenados en su memoria mediante técnicas P2P. Gracias a esta característica, este tipo de modelos constituye una infraestructura natural para poder combinar, de forma eficaz, técnicas colaborativas y basadas en contenido.

Agradecimientos

Este trabajo ha sido financiado parcialmente con fondos provenientes del proyecto 'SAEICS: *Sistema Adaptativo con Etiquetado Inteligente para Correo Spam*' de la Universidad de Vigo. Los autores quieren agradecer los acertados comentarios y sugerencias de los dos revisores anónimos, que sin duda alguna han contribuido profundamente a mejorar la calidad y profundidad de este trabajo.

Referencias

[Ahmed04] S. Ahmed, F. Mithun. 'Word Stemming to Enhance Spam Filtering'. Proceedings of the First Conference on Email and Anti-Spam, http://www.ceas.cc/papers-2004/index.html. (2004).

[AIMC05] Asociación para la Investigación de Medios de Comunicación. http://www.aimc.es/aimc.php. (2005).

[Albrecht et al. 05] K. Albrecht, N. Burri, R. Wattenhofer. 'Spamato – An Extendable Spam Filter System'. Proceedings of the Second Conference on Email and Anti-Spam, http://www.ceas.cc/. (2005).

[Allan et al. 95] J. Allan, L. Ballesteros, P.J. Callan, W.B. Croft, Z. Lu. 'Recent Experiments with INQUERY'. Proceedings of the 4th Text Retrieval Conference, pp. 49-64. (1995).

[Androutsopoulos et al. 00a] I. Androutsopoulos, J. Koutsias, K.V. Chandrinos, G. Paliouras, C. Spyropoulos. 'An evaluation of Naïve Bayesian anti-spam filtering'. Proceedings of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning, pp. 9-17. (2000).

[Androutsopoulos et al. 00b] I. Androutsopoulos, G. Paliouras, V. Karkaletsis, G. Sakkis, C.D. Spyropoulos, P. Stamatopoulos. 'Learning to filter spam e-mail: A comparison of a Naïve Bayesian and a memory based approach'. Proceedings of the Workshop on Machine Learning and Textual Information Access, 4th European Conference on Principles and Practice of Knowledge Discovery in Databases, pp. 1-13. (2000).

[Androutsopoulos et al. 04] I. Androutsopoulos, G. Paliouras, E. Michelakis. 'Learning to Filter Unsolicited Commercial E-Mail'. Informe técnico: TR 2004-2, NCSR National Centre for Scientific Research "Demokritos". (2004).

[AUI05] Asociación de Usuarios de Internet. http://www.aui.es/. (2005).

[Breiman01] L. Breiman. 'Ramdom Forests'. Machine Learning, 45. pp. 2-32. (2001).

[Buckley et al. 94] C. Buckeley, G. Salton, J. Allan, A. Stinghal. 'Automatic Query Expansion Using SMART'. Proceedings of the 3rd Text Retrieval Conference, pp. 69-80. (1994).

[Cavnar94] W.B. Cavnar. 'Using an N-Gram Based Document Representation with a Vector Processing Retrieval Model'. Proceedings of the 3rd Text Retrieval Conference, pp. 269-278. (1994).

[Clayton05] R. Clayton. 'Stopping Outgoing Spam by Examining Incoming Server Logs'. Proceedings of the Second Conference on Email and Anti-Spam, http://www.ceas.cc/. (2005).

[Cmelax05] Nilsimsa. http://ixazon.dynip.com/~cmeclax/nilsimsa.html. (2005).

[Cohen95] W. Cohen. 'Fast effective rule induction'. Proceedings of the 12th International Conference in Machine Learning, pp. 115-123. (1995).

[Cohen99] W. Cohen, Y. Singer. 'Context-sensitive learning methods for text categorization'. ACM Transactions on Information Systems, 17(2). pp. 141-173. (1999).

[Cunningham et al. 03] P. Cunningham, N. Nowlan, S.J. Delany, M. Haahr. 'A Case-Based Approach to Spam Filtering than Can Track Concept Drift'. Proceedings of International Conference on Case Based Reasoning, ICCBR-2003, Workshop of Long-Lived CBR Systems. (2003).

[Deerwester et al. 90] S.C. Deerwester, S.T. Dumais, T.K. Landauer, G.W. Furnas, R.A. Harshman. 'Indexing by Latent Semantic Analysis'. Journal of the American Society of Information Science, 41(6). pp. 391-407. (1990).

[Delany04] S.J. Delany, P. Cunningham. 'An Analysis of Case-Based Editing in a Spam Filtering System'. Proceedings of the 7th European Conference on Case-Based Reasoning, ECCBR-2004, pp. 128-141. (2004).

[Delany et al. 04] S.J. Delany, P. Cunningham, L. Coyle. 'An Assessment of Case-Based Reasoning for Spam Filtering'. Proceedings of the 15th Irish Conference on Artificial Intelligence and Cognitive Science, AICS-2004, pp. 9-18. (2004).

[DesElms05] Spam Links: Everything you didn't to have to know about spam. http://spamlinks.net/. (2005).

[Divmod05] The Divmod spam corpus. http://www.divmod.org/cvs/corpus/spam/. (2005).

[Fdez-Riverola et al. 05] F. Fdez-Riverola, J.R. Méndez, E.L. Iglesias, F. Díaz. 'Representación flexible de e-mails para la construcción de filtros anti-spam: un caso práctico'. Proceedings de las VI Jornadas de Transferencia Tecnológica de Inteligencia Artificial: TTIA 2005, pp. 109-116. (2005).

[Fdez-Riverola et al. 06] F. Fdez-Riverola, E.L. Iglesias, F. Díaz, J.R. Méndez, J.M. Corchado. 'SpamHunting: An Instance-Based Reasoning

System for Spam Labelling and Filtering'. Decision Support Systems. *To appear*. (2006).

[Fdez-Riverola et al. 07] F. Fdez-Riverola, E.L. Iglesias, F. Díaz, J.R. Méndez, J.M. Corchado. 'Applying Lazy Learning Algorithms to Tackle Concept Drift in Spam Filtering'. Expert Systems with Applications, 33(1). *To appear*. (2007).

[Feng et al. 03] H. Feng, O. Kolesnikov, P. Fogla, W. Lee, W. Gong. 'Anomaly detection using call stack information. Proceedings of the IEEE Symposium on Security and Privacy, pp. 62-76. (2003).

[Freund97] Y. Freund, R.E. Schapire. 'A decision-theoretic generalization on on-line learning and an application to boosting'. Journal of Computer and System Sciences, 55(1). pp. 119-139. (1997).

[Fürnkranz94] J. Fürnkranz, G. Widmer. 'Incremental Reduced Error Pruning'. Proceedings of the 12th International Conference on Machine Learning, ICML-1994, pp. 70-77. (1994).

[Gee03] K.R. Gee. 'Using latent semantic indexing to filter spam'. Proceedings of the 2003 ACM symposium on applied computing, pp. 460-464. (2003).

[Golbeck04] J. Golbeck, J. Hendler. 'Reputation Network Analysis for Email Filtering'. Proceedings of the First Conference on Email and Anti-Spam, http://www.ceas.cc/papers-2004/index.html. (2004).

[Gomes et al. 05] H.L. Gomes, R.B. Rodrigo Almeida, L.M.A Bettencourt, V. Almeida, J.M. Almeida. 'Comparative Graph Theoretical Characterization of Networks of Spam and Legitimate Email'. Proceedings of the Second Conference on Email and Anti-Spam, http://www.ceas.cc/. (2005).

[González et al. 05] J.J. González, J.R. Méndez, F. Fdez-Riverola. 'Modelos anti-spam de inteligencia artificial'. Proceedings de la Conferencia Ibero-Americana WWW/Internet 2005, CIAWI-2005, pp. 548-551. (2005).

[Graham02] P. Graham. 'A plan for spam'. http://www.paulgraham.com/spam.html. (2002).

[Graham-Cumming04] J. Graham-Cumming. 'How to beat an adaptive spam filter'. Proceedings of the MIT Spam Conference. http://www.jgc.org/SpamConference011604.pps. (2004).

[Grossman et al. 95] D.A. Grossman, D.O. Holmes, O. Frieder, M.D. Nguyen, C.E. Kingsbury. 'Improving Accuracy and Run-Time Performance for TREC-4'. Proceedings of the 4th Text Retrieval Conference, pp. 433-448. (1995).

[Guenter98] B. Guenter. SPAM Archive http://www.em.ca/~bruceg/spam/. (1998).

[Hettich et al. 98] S. Hettich, C.L. Blake, C.J. Merz. 'UCI Repository of machine learning databases'. http://www.ics.uci.edu/~mlearn/MLRepository.html. Irvine, CA: University of California, Department of Information and Computer Science. (1998).

[Hovold05] J. Hovold. 'Naïve Bayes Spam Filtering Using Word-Position-Based Attributes'. Proceedings of the Second Conference on Email and Anti-Spam, http://www.ceas.cc/. (2005).

[Frostfyre05] I. Frostfyre. 'SpamBlocked'. http://www.spamblocked.com. (2005).

[Jackson02] P. Jackson, I. Moulinier. 'Natural Language Processing for Online Applications: Text Retrieval, Extraction, and Categorization'. Ed. John Benjamins Publishing Co. (2002).

[Jauregi04] A.Z. Jauregi. 'Fundamentos de Latent Semantic Indexing (LSI) y su aplicación a la categorización de textos periodísticos en euskera'. Procesamiento del Lenguaje Natural, 32. pp. 67-74. (2004).

[Joachims98] T. Joachims. 'Text Categorization with Support Vector Machines: Learning with Many Relevant Features'. Proceedings of the 10th European Conference on Machine Learning, ECML-1998, pp. 137-142. (1998).

[Joachims97] T. Joachims. 'A probabilistic analysis of the Rochio algorithm with TFIDF for text categorization'. Proceedings of the 14th International Conference on Machine Learning, ICML-1997, pp. 143-151. (1997).

[John95] G. John, P. Langley. 'Estimating continuous distributions in Bayesian classifiers'. Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence, pp. 338-345. (1995).

[Judge02] P. Judge. 'Spam Archive: Donate Spam to Science'. http://www.spamarchive.org/. (2002).

[Kinley05] A. Kinley. 'Acquiring Similarity Cases for Classification Problems'. Proceedings of the 6th International Conference on Case-Based Reasoning, ICCBR-2005, pp. 327-338. (2005).

[Kohavi95] R. Kohavi. 'A study of cross-validation and bootstrap for accuracy estimation and model selection'. Proceedings of the 14th International Joint Conference on Artificial Intelligence, IJCAI-1995, pp. 1137-1143. (1995).

[Lee05] H. Lee, A. Ng. 'Spam Deobfuscation using a Hidden Markov Model'. Proceedings of the Second Conference on Email and Anti-Spam, http://www.ceas.cc/. (2005).

[Leiba04] B. Leiba, N. Borenstein. 'A Multifaceted Approach to Spam Reduction'. Proceedings of the First Conference on Email and Anti-Spam, http://www.ceas.cc/papers-2004/index.html. (2004)

[Leiba et al. 05] B. Leiba, J. Ossher, V.T. Rajan, R. Segal, M. Wegman. 'SMTP Path Analysis'. Proceedings of the Second Conference on Email and Anti-Spam, http://www.ceas.cc/. (2005).

[Lenz96] M. Lenz, H.-D. Burkhard. 'Case Retrieval Nets: Foundations, properties, implementation, and results'. Informe técnico: Humboldt University, Berlin. (1996).

[Lowd05] D. Lowd, C. Meek. 'Good Word Attacks on Statistical Spam Filters'. Proceedings of the Second Conference on Email and Anti-Spam, http://www.ceas.cc/. (2005).

[Mason05] J. Mason. The Apache SpamAssassin Public Corpus. http://spamassassin.apache.org/publiccorpus/. (2005).

[Méndez et al. 05] J.R. Méndez, E.L. Iglesias, F. Fdez-Riverola, F. Díaz, J.M. Corchado. 'A Comparative Impact Study of Corpus Preprocessing for the Construction of Anti-Spam Filtering Software'. Proceedings de la XI Conferencia de la Asociación Española para la Inteligencia Artificial, pp. 29-38. (2005).

[Méndez et al. 06] J.R. Méndez, F. Fdez-Riverola, E.L. Iglesias, F. Díaz, J.M. Corchado. 'Tracking Concept Drift at Feature Selection Stage in SpamHunting: an Anti-Spam Instance-Based Reasoning System'. Proceedings of the 8th European Conference on Case-Based Reasoning: ECCBR-2006, pp. 504-518. (2006).

[Michelakis et al. 04] E. Michelakis, I. Androutsopoulos, G. Paliouras, G. Sakkis, G.P. Stamatopoulos. 'Filtron: A Learning-Based Anti-Spam Filter'. Proceedings of the First Conference on Email and Anti-Spam, http://www.ceas.cc/papers-2004/index.html. (2004).

[Moustakas et al. 05] E. Moustakas, C. Ranganathan, P. Duquenoy. 'Combating spam through legislation: a comparative analysis of US and European approaches'. Proceedings of the Second Conference on Email and Anti-Spam, http://www.ceas.cc/. (2005).

[Mueller05] S.H. Mueller. Spam Abuse Network. http://spam.abuse.net. (2005).

[Orasan02] C. Orasan, R. Krishnamurthy. 'The Junk-Email Corpus'. http://clg.wlv.ac.uk/projects/junk-email/corpus-no-duplications.tar.gz. (2002).

[Porter80] M. Porter. 'An algorithm for suffix stripping'. Program, 14(3). pp. 130-137. (1980).

[Prakash05] V.V. Prakash. 'Vipul's Razor: Home'. http://razor.sourceforge.net/. (2005).

[Provost99] J. Provost. 'Naïve-bayes vs. rule-learning in classification of email'. Informe técnico. Dept. of Computer Sciences at the University of Texas at Austin. (1999).

[Pyzor05] Pyzor. http://pyzor.sourceforge.net/. (2005).

[Quinlan93] J.R. Quinlan. 'C4.5: programs for machine learning'. Ed. Morgan Kaufmann Publishers Inc. (1993).

[Rhyolite00] Rhyolite Software. 'DCC: Distributed Checksum Clearinghouse'. http://www.rhyolite.com/anti-spam/dcc/. (2000).

[Rigoutsos98] I. Rigoutsos, A. Floratos. 'Combinatorial Pattern Discovery in Biological Sequences: the TEIRESIAS Algorithm'. Bioinformatics. 14(1). pp. 55-67. (1998).

[Rigoutsos04] I. Rigoutsos, T. Huynh. 'Chung-Kwey: a Pattern-discovery-based System for the Automatic Identification of Unsolicited E-mail Messages (Spam)'. Proceedings of the First Conference on Email and Anti-Spam, http://www.ceas.cc/papers-2004/index.html. (2004).

[Rijsbergen79] K. Rijsbergen. 'Information Retrieval'. Ed. Butterworth. (1979).

[Rios04] G. Rios, H. Zha. 'Exploring Support Vector Machines and Random Forests for Spam

Detection'. Proceedings of the First Conference on Email and Anti-Spam, http://www.ceas.cc/papers-2004/index.html. (2004).

[Rocchio71] J. Rocchio. 'Relevance Feedback in Information Retrieval'. En G. Salton: *The SMART Retrieval System: Experiments in Automatic Document Processing*, Chapter 14. pp. 313-323. Ed. Prentice Hall. (1971).

[Segal04] R. Segal, J. Crawford, J. Kephart, B. Leiba. 'SpamGuru: An Enterprise Anti-Spam Filtering System'. Proceedings of the First Conference on Email and Anti-Spam, http://www.ceas.cc/papers-2004/index.html. (2004).

[Smeaton et al. 94] A.F. Smeaton, F. Kelledy, R. O'donnel. 'Indexing Structures Derived from Syntax'. Proceedings of the 3rd Text Retrieval Conference, pp. 55-68. (1994).

[SpamHaus05a] SpamHaus, 'Definition of Spam'. http://www.spamhaus.org/definition.html. (2005)

[SpamHaus05b] SpamHaus Project. http://www.spamhaus.org. (2005).

[SpammerX05] SpammerX. 'Spam'. Ed. Anaya. (2005).

[Stanley03] K.O. Stanley. 'Learning concept drift with a committee of decision trees'. Informe técnico: UT-AI-TR-03-302, Department of Computer Sciences, University of Texas at Austin, USA. (2003).

[SuretyMail05]. Surety Mail. 'ISIPP IADB Email Senders Accreditation Program'. http://www.isipp.com/iadb.php. (2005).

[Taylor04] G. Taylor. The Grant Taylor Spam Email Corpus. http://www2.picante.com:81/~gtaylor/download/spam.tar.gz. (2004).

[TheRegister05] The Register. http://www.theregister.co.uk. (2005).

[TrendMicroInc05]. Trend Micro Incorporated. 'MAPS'. (2005).

[Tsymbal04] A. Tsymbal. 'The problem of concept drift: definitions and related work'. Informe técnico: TCD-CS-2004-15, Departament of Computer Science Trinity College, Dublin, https://www.cs.tcd.ie/publications/tech-reports/reports.04/TCD-CS-2004-15.pdf. (2004).

[Vapnik99] V. Vapnik. 'The nature of Statistical Learning Theory'. 2nd Edition Statistic for Engineering and Information Science. Ed. Springer. (1999).

[Wayne92] I. Wayne, P. Langley. 'Induction of One-Level Decision Trees'. Proceedings of the Ninth International Workshop on Machine Learning: ML-1992, pp. 233-240. (1992).

[Widmer03] G. Widmer, M. Kubat. 'Effective learning in dynamic environments by explicit context tracking'. Proceedings of the 6th European Conference on Machine Learning, ECML-1993, pp. 227-243. (2003).

[Wittel04] G.L. Wittel, S.F. Wu. 'On Attacking Statistical Spam Filters'. Proceedings of the First Conference on Email and Anti-Spam, http://www.ceas.cc/papers-2004/index.html. (2004).

[Zhang et al. 04] L. Zhang, J. Zhu, T. Yao. 'An Evaluation of Statistical Spam Filtering Techniques'. ACM Transactions on Asian Language Information Processing, 3(4). pp. 243-269. (2004).